## **Data visualization and Text mining**

## PROF. ANDREA BELLI

***COURSE AIMS AND INTENDED LEARNING OUTCOMES***

This course focuses on data visualization and text processing techniques, providing theorical context and practical experience on tools and algorithm to discover patterns, extract knowledge, and support decision making.

***Objectives***

At the end of the course, students are expected to be able to:

***Knowledge and understanding***

* understand the basic concepts and principles of data visualization;
* know the standards and tools that represent the state of the art of data visualization, in the field of data science;
* understand the basic concepts, principles, and major algorithms in text mining;
* learn the main natural language processing approaches.

***Applying knowledge and understanding***

* analyze and manipulate information and complex data in order to process clear and accurate summary;
* build machine learning and LSTM models for text classification, named entity extraction and sentiment analysis;
* implement text processing pipelines Python, and tools such as Jupyter Notebook or Google Colab.

***Making judgements***

* recognize the features of a dataset and choose the data visualization that best represents them;

***Communication skills***

* use graphics and interaction methods to correctly communicate the context information of a data visualization;
* argue their ideas and conclusions by comparing them both with those of their group mates within their team and outside.

***Learning skills***

* be able to independently consult the developer documentation of a library in any programming language
* apply an iterative approach, in which practical experimentation reinforces the concepts learned during the theoretical lessons;
* develop creativity, considering the error and the unexpected as a basis for future experimentation.

***COURSE CONTENT***

***Module A***

1. Introduction to data visualization and open data with examples of real projects.

2. Data formats, web standards, libraries and rules for good code.

3. Practical principles of data visualization: linking data dimensions to visual variables through scales (encoding).

4. Dataset uploading, data processing and standard visualizations trough Altair Python library

5. Roles and functions of data visualization: comparison, correlation, ranking, distribution, part to whole, flow, time, space, hierarchy.

6. Style of visualization: theme, palette and font.

7. Enrich the data with information: colors, graphics and visual guides (titles, axes, formats, legends)

***Module B***

1. Natural Language Processing: features, state of the art of open source and reference market, Text Preprocessing and Representation; Part-of-Speech tagging, Dependency tree generation, Entity Recognition, Sentence Segmentation.

2. Machine Learning techniques applied to Text Processing; Classification Metrics, Text Extraction features identification, practice with Scikit-Learn package on real dataset.

3. Word Similarity, Word-2-Vec usage and generation, Sentiment Analysis concepts and available solutions.

4. Deep Learning basics applied to Text; Understanding of RNN, LSTM and GRU.

5. Transformers, Attention algorithm, GPT and BERT.

***READING LIST***

J. Laumans, An introduction to Visualizing Data, online

J. Dougherty-I. Ilyankou (2021), Hands-On Data Visualization, O’reilly and online at handsondataviz.org

Documentation of Altair Python library: online at altair-viz.github.io

Documentation of Vega Lite Javascript library: online at vega.github.io/vega-lite/docs/

C. Ritchie-C. Mellish (2000), Techniques in Natural Language Processing.

T. Mikolov-Kai Chen-G. Corrado-J. Dean, Efficient Estimation of Word Representations in Vector Space.

J. Devlin-Ming-Wei Chang-K. Lee-K. Toutanova Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding.

J. Howard-S. Ruder Universal Language Model Fine-tuning for Text Classification.

***TEACHING METHOD***

The course is structured as follows:

* The course is held twice a week based on the academic calendar.
* Students are required to participate actively to the lessons.
* Teaching support includes slides, case studies and exercises; a notebook is recommended for running the exercises during the lessons.

***ASSESSMENT METHOD AND CRITERIA***

Attendants will be asked to develop a project for Module A and B; the project is then discussed during an oral interview that involves questions about theory, methods, coding and interpretation of the results.

***NOTES AND PREREQUISITES***

Correspondence between students and teachers will be managed through Blackboard and by email.

Students enrolling in this course should have

* a basic understanding of Python development
* a primary knowledge of formats and tools for data storage